CLAIMS:

I claim:

1. A true random number generator comprising:
   a receiver to receive a signal comprising a predetermined source data;
   a recovery circuit to recover data from the received signal;
   a controller to sufficiently stress the recovery circuit such that at least a portion of the recovered data differs from respective portions of the predetermined source data;
   an extractor to define a random number based upon differences between the recovered data and the predetermined source data.

2. The true random number generator of claim 1, in which:
   the receiver and the recovery circuit comprise a clock recovery circuit; and
   the controller is operable to influence at least one of the data transfer rates of the signal directed to the receiver, the number of sequential same-state data bits of the predetermined source data, and the stability of the clock recovery circuit for establishing the stress.

3. The true random number generator of claim 2, in which the extractor comprises:
   a comparator to compare and determine errors between the recovered data and reference data related to the predetermined source data;
   a counter to count errors determined by the comparator; and
   a sampler to sample at least a portion of the bits of the counter to define the random number.

4. The true random number generator of claim 3, further comprising:
   a first memory to source the predetermined source data;

a data transmitter to receive the predetermined source data from the first memory and output the signal comprising the predetermined source data to the receiver; and

second memory comprising the reference data to be supplied to the comparator, the reference data substantially the same as the predetermined source data.

5. The true random number generator of claim 4, wherein the data transmitter and the receiver form part of a multi-gigabit data transceiver embedded within a programmable logic device, the programmable logic device comprising a configurable link coupled between the receiver and the transmitter.

6. The true random number generator of claim 1, wherein the extractor defines the random number based on an interval of time required to reach a predetermined number of differences between the recovered data and the predetermined source data.

7. A true random number generator comprising:

a transceiver;

a jitter performance tester to exercise the transceiver with predetermined data;

the jitter performance tester pre-configured to stress the operability of the transceiver for increasing an error probability of the transceiver;

extraction circuit to obtain a random number sequence based on differences between the data resolved by the transceiver and the predetermined data.

8. The random number generator of claim 7, the jitter performance tester to control at least one of the data of the predetermined data and an associated data transfer rate to influence the error probability.

9. The true random number generator of claim 7, in which:

the transceiver comprises a receiver to receive and recover data from a data signal comprising the predetermined data, and to format the recovered data into data words;

the jitter performance tester comprises:

a comparator to compare and determine differences between the recovered data words and respective data words associated with the predetermined data; and

a counter to count a number of differences determined by the comparator; and

the extractor to determine numbers for the random number sequence based upon counts determined by the counter over respective durations of the predetermined data.

10. The true random number generator of claim 9, in which the duration associated with each count encompasses a plurality of comparisons performed by the comparator.

11. The true random number generator of claim 9, in which the extractor further comprises a sampler to sample at least a portion of bits of the counter.

12. The true random number generator of claim 11, the extractor to enable the sampler once every counter duration.

13. The true random number generator of claim 12, the extractor operable to control length of the durations.

14. The true random number generator of claim 9, in which the transceiver, the jitter performance tester, and the extractor are embedded within a programmable logic device.

15. The true random number generator of claim 14, further comprising first memory to source the predetermined data, and second memory to source the respective data words of the predetermined data to the comparator.

16. The true random number generator of claim 15, in which the transceiver further comprises a transmitter to obtain data from the first memory and output it to the receiver.

17. The true random number generator of claim 16, further comprising a RS-232 interface to sample at least a portion of the counter, with a sampling rate less than a data transfer rate of the transmitter.

18. The true random number generator of claim 9, in which the extractor is operable to define the random numbers based upon a number of bit differences determined by the comparator.

19. The true random number generator of claim 9, in which the extractor is operable to define the random numbers based upon the respective durations required to produce a predetermined number of difference counts as determined by the comparator and the counter.

20. A method of generating a random number, comprising:
comparing first data to reference data and determining differences therebetween;
counting the differences determined;
performing the comparing, determining and counting over a duration; and
defining a random number based on at least one of the differences counted and the duration associated with the counting.

21. The method of claim 20, further comprising, relating the first data to the reference data with a given error probability.

22. The method of claim 20, further comprising:
recovering data from a data signal; and
using the recovered data for the first data of the comparing;
the data signal comprising data substantially the same as the reference data; and
the recovering comprising an error probability greater than zero and less than 1.

23. The method of claim 22, further comprising influencing jitter performance of the data recovery.

24. The method of claim 23, in which the influencing of the jitter performance comprises establishing at least one of the sequences of data for the data signal, the data transfer rate, and the stability of a clock recovery process associated with the data recovery.

25. The method of claim 24, in which run length for a sequence of same state data for the data is configured for a duration sufficient to reach a waterfall region of a jitter curve characteristic of the clock recovery process.

26. The method of claim 22, further comprising:
formatting the recovered data for the first data into word format;
the comparing and counting to comprise:
comparing words of the reformatted recovered data to words of the reference data; and
counting the number of bit errors therebetween.

27. The method of claim 22, further comprising:
formatting the recovered data for the first data into parallel format;
the comparing and the counting to comprise:

comparing words of the reformatted data relative to respective words of the reference data to determine any differences therebetween; and

counting a number of comparisons yielding a difference determination.

28. The method of claim 27, further comprising continuing the counting for duration to encompass multiple word-to-word comparisons for each random number defined.

29. The method of claim 28, further comprising defining the random number as least significant bits of the number counted.

30. The method of claim 29, further comprising:

encrypting a communication signal using a seed based encryption key;

forming the encryption key with seed values based on the defined random numbers.

31. The method of claim 22, further comprising:

configuring predetermined source data within a first memory to comprise a stress sequence of same-state data;

defining a run-length for the stress sequence of same-state data to extend over a stress duration;

retrieving the predetermined source data from the first memory device;

multiplexing words of the source data retrieved from the first memory to convert it from a parallel format into a serial format;

transmitting the serially formatted data;

receiving the transmitted serially formatted data as the data signal;

retrieving the reference data related to the predetermined source data from a second memory; and

storing the stress sequence as at least a part of the reference data in the second memory.

32. The method of claim 31, further comprising:

storing conditioning data in the first memory as a preamble before the stress sequence;

when receiving the preamble, using transitions of the conditioning data to synchronize a recovered clock; and

after the synchronizing of the recovered clock with the preamble and during an interval of time associated with receipt of the transmitted stress sequence, performing the retrieval of the reference data, the comparing, and the determining and the counting of errors.

33. The method of claim 32, further comprising repeating each of:

the retrieving, the multiplexing and the transmitting to again transmit a preamble and stress sequence;

the receiving of the transmitted, serially-formatted data corresponding to the preamble and stress sequence;

the retrieving of the reference data, the comparing and the determining and counting of errors; and

the counting to accumulate respective counts of the determined differences.

34. The method of claim 33, further comprising continuing the repeating and count accumulations through the count duration associated with the random number to be defined.

35. The method of claim 34, further comprising:

determining a time lapse for a predefined error probability or jitter characteristic curve associated with clock recovery of magnitude sufficient to reach a waterfall region of the characteristic curve; and

defining the run-length of the stress sequence based on the determined time lapse.

36. The method of claim 35, further comprising sampling the accumulated count after a plurality of repeats.